

Development of a High Performance Computer cluster to perform parallel computation: case study

Mohammed Rumman Ahmed¹, Yasmeen Haque¹,
Mohammad Shahidur Rahman²

Syed Boduzzaman Faruque¹, Md Enamul Haque¹, Sarfuddin Tarek¹

¹*Department of Physics, Shahjalal University of Science and Technology, Sylhet, Bangladesh,*

²*Department of Computer Science & Engineering, Shahjalal University of Science and Technology, Sylhet.*
Corresponding Author: rummann@hotmail.com

Abstract

This paper presents the design and development of a high-performance computer cluster that can perform parallel computational task in batch mode. In order to benchmark the performance of the developed system, we have studied two cases related to the intense computational task. Firstly, we performed a simulation using Gaussian09 package with time dependent density functional theory at B3LYP/6-31G level to find out the UV-vis Spectrum of L-Tyrosine by increasing the number of compute nodes. The absorption spectra were also compared with the experimentally obtained results from the UV-vis spectrometer. Secondly, we dealt with a three-body problem by solving the Sitnikov equation of motion.

Keywords: *High Performance Computing, Computer Cluster, Parallel Computing, Gaussian09 Simulation, UV-VIS Spectroscopy, L-Tyrosine, Sitnikov solution.*

Date of Submission: 13-12-2021

Date of Acceptance: 28-12-2021

I. Introduction

For last centuries, Science has advanced through a combination of theory and experiment. In recent decades, a third and powerful new approach has emerged, namely Computational Science[1]. Computational Science use computers to simulate the behavior of a system, rather than observing the system or building a physical model of it. The simulation processes are basically the solution of some large and or complex mathematical models describing the system.

Most of the scientific simulation requires massive computational workload. Now a day's a single powerful desktop computer usually fails to take this workload. One way of overcoming this situation is to run the job on any costly commercial supercomputer. Another alternative of this problem is to break down the job to many smaller pieces and distribute the workload to a number of computers connected together[2] through local network which is very cost effective. This type of computer system which is an optimized combination of computers, interconnect network and Software for parallel task is called High Performance Computer Cluster or simply HPC [3].

Now a day's as simulation is being used in almost every field of study of science; from the molecular systems to celestial system, HPC became the essential tool for simulation method and useful in all computationally intensive research areas including physics, chemistry, statistics, bio-informatics, etc.

Over the past decades many software packages have been developed for simulation of complex molecular systems. To study Geometry optimization, Single point energy, Frequencies and Thermochemistry, Uv-Vis and Electronic Transitions, one of the most popular Molecular Mechanic Simulation package is Gaussain™ [4] . It is released in 1970 by a research group at Carnegie-Mellon University and has been continuously updated since then.

Gaussain™ is the most widely used program package for performing electronic structure calculations for its performance and accuracy. It utilizes fundamental laws of quantum mechanics using a specific set of approximation method. These approximations are combined with a calculation algorithm and are applied to atomic orbitals, defined by the basis set (21g, 6-31g, 6-311g(d,p), cc-pvdz, aug-cc-pvtz , etc are a set of wave functions that describes the shape of atomic orbitals) [5]. In general, the methods are separated into 4 main types: semi-empirical,(am1, pm3, mndo, etc.), Ab-initio(hf, mp2, ccsd, ccsdt, etc.), Density functional theory (b3lyp, mpw1pw91, etc) and – Hybrid (G2, G3, ...) [6]. Higher levels of method are often more accurate and take massive computational power and time.

In computational physics in the field of celestial mechanics Sitnikov problem[7]. is one of the nonintegrable problems that attempts to describe the movement of three celestial bodies due to their mutual gravitational attraction. It has been studied by many authors, using various mathematical methods. In 1913, MacMillan (1913) presented a particular case of this problem now known as the MacMillan or the circular Sitnikov problem in which the primaries describe circular orbits. This problem is integrable via quadrature involving elliptic integrals. Belbruno et al. (1994) gave analytic expressions for the solutions of the MacMillan problem using elliptic functions.

Recently, Faruque (2003)[8] discussed analytic approximations to the MacMillan problem. The elliptic Sitnikov problem, or simply the Sitnikov problem, became important when Sitnikov (1960) presented the problem to show the existence of oscillatory motions in the three-body problem. To our knowledge, a closed form solution of the Sitnikov problem was first found by Hagel (1992)[9]. Using perturbation theory, Hagel (1992) first obtained a solution of the linearized equation of motion and then extended the problem by including nonlinear forces and succeeded in writing a closed form solution. This solution of Hagel (1992) is applicable for moderate primary eccentricities for small oscillation around barycentre.

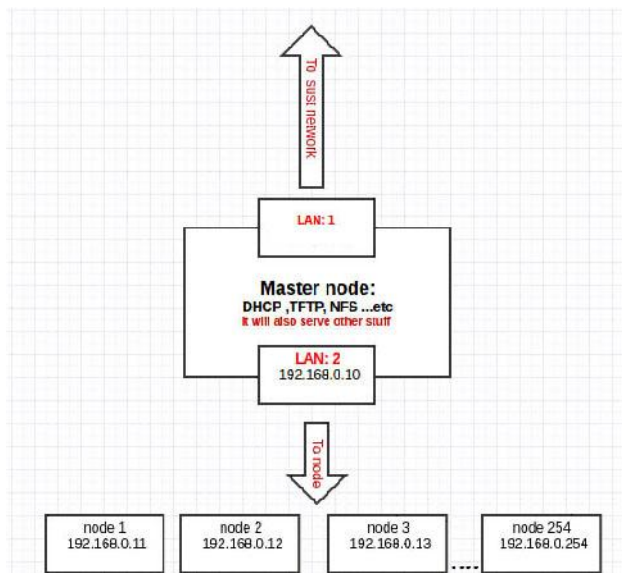
In order to study the described cases, we have developed an HPC Cluster with commonly available computer hardware and configured appropriately based on the Linux Operating System With the help of this concerned HPC, we conducted a simulation for UV-vis spectrum of a complex bio-molecule L-Tyrosine. We have also analyzed motion of a celestial three-body system on the developed HPC.

In *Section 2* of this paper a brief description has been presented about the HPC cluster that has been build and deployed in the NLO lab of Department of Physics, SUST. In *Section 3* we describe a simulation of UV-vis spectrum of Tyrosine molecule in our newly build HPC cluster as a parallel computational task and comparison with experimental result. *Section 4* contains the regenerated graph of the solution of Sitnikov three body problems as a remote serial job of the HPC.

II. HPC DEVELOPMENT:

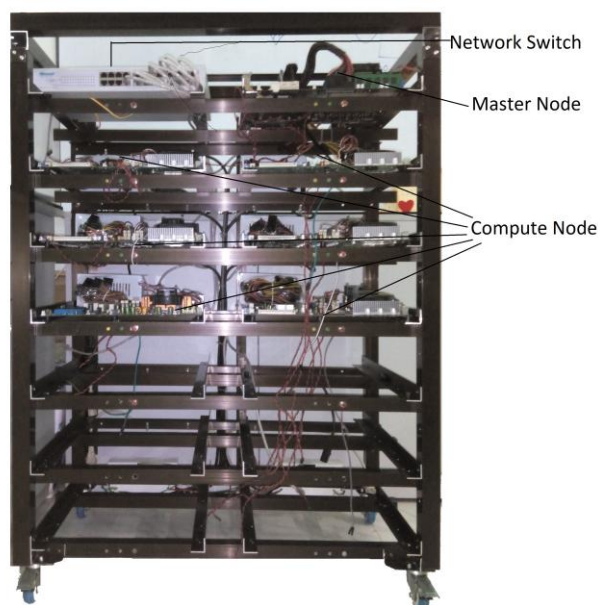
The developed HPC consist of one Master node and 8 disk less Computing nodes that are connected through a high-speed network switch. A details of the HPC configuration is provided in Appendix A. This HPC has the ability to perform any required large computation by scaling it to 250 Computing nodes due to a Class-C network IP configuration (Fig -1). We have used a server rack to mount all the hardware and required accessories.

Fig1 : Image of the computer cluster system



In our case, we have used star topology of network for the Computing nodes centering master node to achieve efficient communication. Using 1024 mbps Ethernet interface, all the nodes have been connected through a Gigabit switch and Cat-6 cable. A public IP is assigned to for the remote access of the Master node A block diagram of the communication is shown in *Fig-1*

Fig2: Image of the computer cluster system



A popular Linux distribution Ubuntu (64-bit) is used as the Operating System by configuring the DHCP[10], TFTP[11], PXELINUX[12], NFS[13] services to achieve basic HPC system. We have ensured that the Message Passing Interface - MPI [14] works appropriately for the basic setup. To enhance the usability, we have implemented the Job scheduler – SLURM [15], User Management tools, basic security protocol at <http://yzhpc.sust.edu>.

In the next sections we describe shortly about the simulation that has been done to test the HPC Cluster.

III. L-TYROSINE AS CASE STUDY 1:

UV-vis absorption spectrum of L-Tyrosine molecule has been simulated on the newly built computer cluster. A solutions of L-Tyrosine has been prepared at different concentration to determine the UV-VIS absorption spectrum of Tyrosine molecule experimentally by Shimadzu UV-1800 Spectro-photometer. Both the theoretical and experimental results have been compared.

3.1 SIMULATION ON HPC: Theoretical prediction of UV-vis Spectrum of L-Tyrosine

Gaussian09 package with Linda parallel library has been used for the simulation on the cluster computer. The Gaussian job type was to calculate the energy up to 10 excited states at DFT level using cc-pvtz basis set with ucam-b3lyp correlation coefficient. To submit and run a Gaussian job in this cluster computer two input files are required. The first one is a *.com* Gaussian input file another one is *.sh* job scheduler file as described in Appendix-B

As the theoretical UV-vis calculation require intensive computational processing power, along with the simulation to analyze the performance of the cluster computer, varying number of Compute-Node and CPU-core has been used to find out the time involved to process the same job. Time around data has been taken as in **Table 5.2**

Optimized output of Tyrosine molecule after the calculation viewed by Gausview is given below in **Fig: 3.1** and simulated output for UV-vis spectrum is given in **Fig: 5.1**

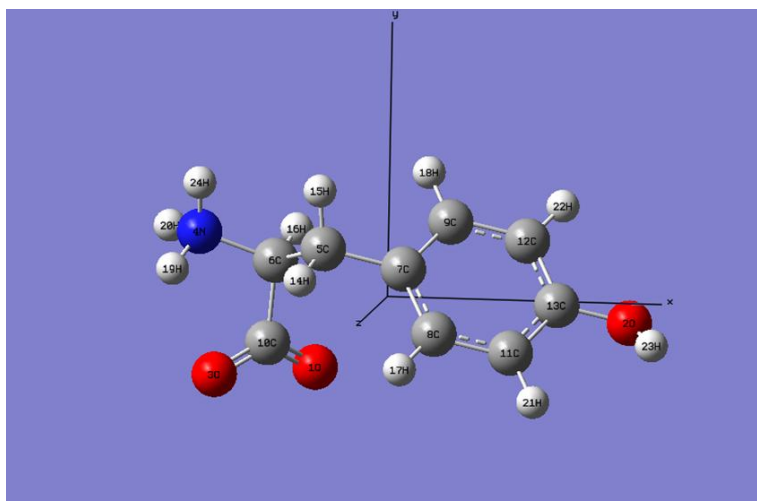


Fig 3.1: Optimized form of Tyrosine -Gaussian output file viewed with Gaussview5

3.2 EXPERIMENT: UV-Vis Spectroscopy Of L-Tyrosine

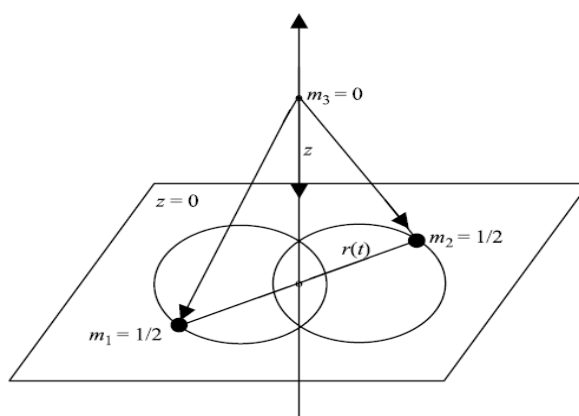
As a material of interest in simulation study L-Tyrosine was also studied for its electronic structure by UV-visible Spectroscopy [16]. Absorbance of L-Tyrosine at different concentrations have been investigated by Shimadzu UV-1800 Spectro-photometer in the range of 190 to 400 nm. Data have been collected at medium speed with sampling interval of 0.5 nm and finally analyzed with UV-ProbeTM . The results of UV-Vis spectroscopy are given in Fig 5.2. [17].

IV. SOLUTION OF SITNIKOV MOTION AS CASE STUDY 2:

Several graph has been generated from the approximate solution of the sitnikov problem for low eccentricities using Octave language as a single node remote job on HPC

In the Sitnikov Problem, two equal primary masses of mass m move on two coplanar elliptic orbits of eccentricity e , $0 \leq e \leq 1$ around their center of mass while a third body of infinitesimal mass m_3 moves on a line that goes through the center of mass of the primaries and is perpendicular to the plane of motion of the primaries.

Fig 4.1: Geometry of Sitnikov Problem



Consider the system consists of two primary bodies of equal mass, and the unit of mass, length, and time in such a way that $m = 1/2$, the gravitational constant $G = 1$, the period of the orbit described by the primaries is 2π . In such a system that the third body of infinitesimal mass m_3 only moves in one dimension – it moves only along the z-axis.

Now, if z denotes the position of the planetoid in a coordinate system with the origin at the centre of mass of the primaries and orbital plane of the primaries to be the x-y plane, then the equation of motion for the third body in Sitnikov problem is represented by the equation

$$\ddot{z} = -\frac{z}{[z^2 + r^2(t)]^{3/2}}, \quad \dots \quad \dots \quad \dots \quad (4.1)$$

where $r(t)$ is the distance of a primary to their center of mass. Figure 4-1 illustrates the geometry of the problem. The Approximate first order Analytical Solution by Faruque, S. B [8] is given below

$$\begin{aligned} z = & \frac{z_0(1 - (24/31)e \cos t)}{1 - (24/31)e} \left[\left(1 + \frac{3}{16}z_0^2 \right) \cos \times \right. \\ & \times \left(\omega t + \frac{48}{31}e\omega \sin t \right) - \frac{3}{16}z_0^2 \cos \left(3\omega t + \frac{144}{31}e\omega \sin t \right) + \\ & + \beta e \left(1 - \frac{1}{2} \cos \left(2\omega t + \frac{96}{31}e\omega \sin t \right) \right) \times \\ & \times \left(\frac{5z_0^2\sqrt{8}}{32\{(3\omega + 1)^2 - \gamma\}} \cos \xi_1 + \frac{5z_0^2\sqrt{8}}{32\{(3\omega - 1)^2 - \gamma\}} \cos \xi_2 + \right. \\ & + \frac{3z_0^2\sqrt{8}}{32\{(5\omega + 1)^2 - \gamma\}} \cos \xi_3 + \\ & + \left. \frac{3z_0^2\sqrt{8}}{32\{(5\omega - 1)^2 - \gamma\}} \cos \xi_4 - \right. \\ & - \frac{5z_0^2\sqrt{8}}{32} \left\{ \frac{1}{(3\omega + 1)^2 - \gamma} + \frac{1}{(3\omega - 1)^2 - \gamma} \right\} - \\ & \left. - \frac{3z_0^2\sqrt{8}}{32} \left\{ \frac{1}{(5\omega + 1)^2 - \gamma} + \frac{1}{(5\omega - 1)^2 - \gamma} \right\} \right], \end{aligned}$$

... .. (4.2)

where

$$\begin{aligned} \xi_1 = & (3\omega + 1)t + \frac{48}{31}e \sin t + \frac{144}{31}e\omega \sin t + \\ & + \frac{3\omega + 1}{2\omega} \sin \left(2\omega t + \frac{96}{31}e\omega \sin t \right), \\ \xi_2 = & (3\omega - 1)t - \frac{48}{31}e \sin t + \frac{144}{31}e\omega \sin t + \\ & + \frac{3\omega - 1}{2\omega} \sin \left(2\omega t + \frac{96}{31}e\omega \sin t \right), \\ \xi_3 = & (5\omega + 1)t + \frac{48}{31}e \sin t + \frac{240}{31}e\omega \sin t + \\ & + \frac{5\omega + 1}{2\omega} \sin \left(2\omega t + \frac{96}{31}e\omega \sin t \right), \\ \xi_4 = & (5\omega - 1)t - \frac{48}{31}e \sin t + \frac{240}{31}e\omega \sin t + \\ & + \frac{5\omega - 1}{2\omega} \sin \left(2\omega t + \frac{96}{31}e\omega \sin t \right). \end{aligned}$$

... .. (4.3)

where,

$$\begin{aligned} \gamma &= \omega_r^2 - \frac{3}{2}\alpha A^2, & \omega_r &= \sqrt{8}, & A &= z_0\sqrt[4]{8}. \\ \delta &= \frac{3}{2}\alpha A^2. & \alpha &= 6\sqrt{8}, & & \\ & & \beta &= \frac{66}{31}\sqrt{8}. & & \end{aligned}$$

And the plot of the equation 4.2 according to Faruque, S. B, is given below Fig-4.2 which is strikingly similar in their variation of amplitude with Hagel (1992)

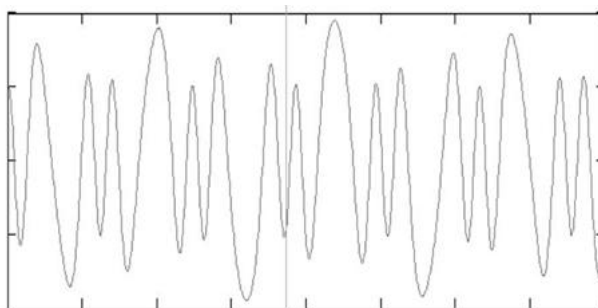


Figure 4.2: Graph of Sitnikov motion for $z_0 = 0.2$ and $e = 0.4$.

To re-generate graph of Sitnikov motion, of the equation 4.2 with our HPC an open source programming language OCTAVE has been used. Necessary script (**Appendix C**) has been written to generate the plot and then run on cluster through octave interpreter. Generated output graph for several Position of z_0 and eccentricity e , is given in **Fig: 5.3**, **Fig: 5.4** and **Fig: 5.5**.

V. RESULTS

CASE STUDY 1:

(i) Simulated UV-vis result form HPC with Gaussian09 For L-Tyrosine Molecule

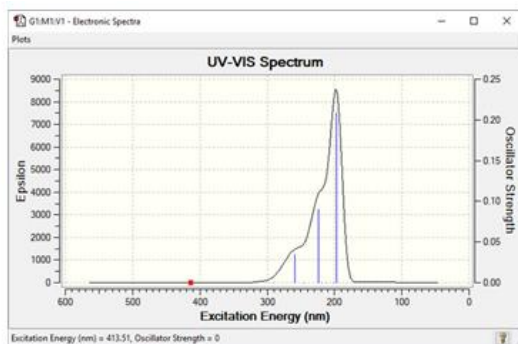


Fig 5.1 Gaussian09 generated output graph of UV- vis Spectrum for our input file

(ii) Experimental result from UV-vis spectrometer For L-Tyrosine Molecule:

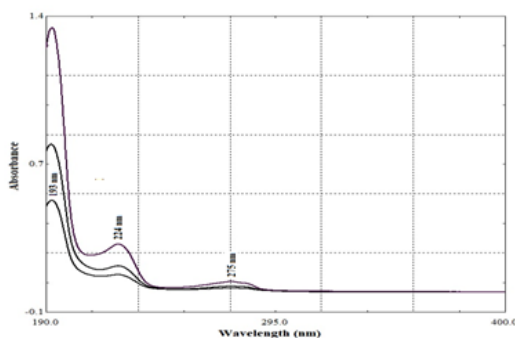


Fig 5.2: UV-vis spectroscopic graph of L- Tyrosine from UV-probe

Observed absorption peak of L-Tyrosine molecule from Simulation and Experiment are summarized in the **Table 5.1** below:

<i>Peaks</i>	<i>Simulated Value of Absorption peaks</i>	<i>Experimental Value of Absorption peaks</i>
1	198nm	193nm
2	225nm	224nm
3	260nm	275nm

Table 5.1 Comparison between simulated and experimental UV-vis absorption of L-Tyrosine

(iii) Time around data for increasing number of CPU-Cores vs Run-Time for the same job:

<i>Task</i>	<i>Compute-Nodes & CPU-Cores</i>	<i>Run-Time in seconds</i>
First Run	1 node or 4 cores,	63217 (17.56 hour)
Second Run	2 nodes or 8 cores	37620 (10.45 hour)
Third Run	3 nodes or 12 cores	19391 (5.38 hour)
Fourth Run	4 nodes or 16 cores	15228 (4.23 hour)

Table 5.2 : Compute-Node VS Job completion time

And, the graph from above table for Time around data for increasing number of CPU-Cores vs Run Time :

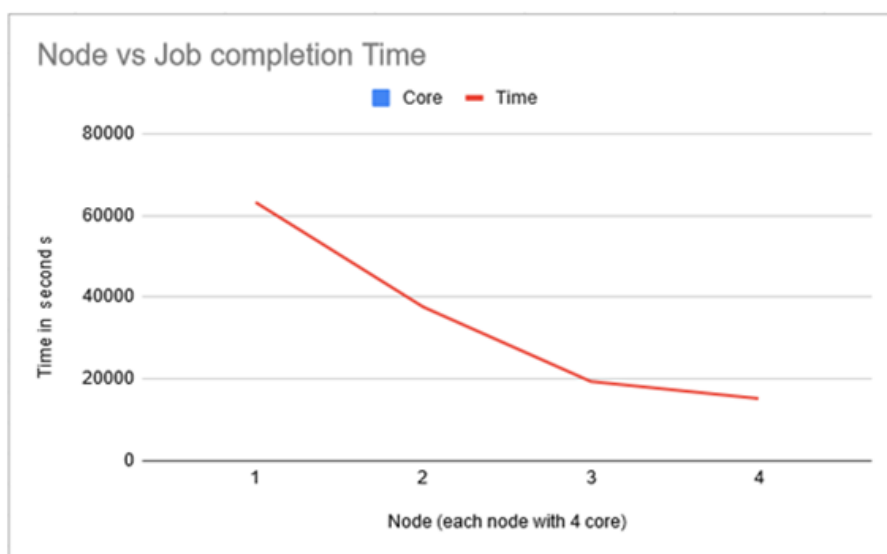


Fig: 5.3 Compute-Node vs Job Completion Time Graph

CASE STUDY 2:

Regenerated graph of Sitnikov motion for different values of z_0 and e with the HPC where, z_0 along x axis and time t along y axis:

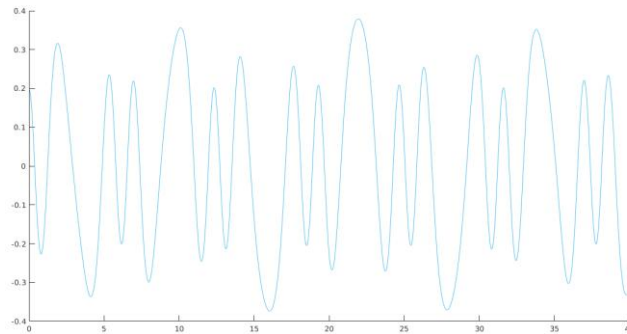


Fig 5.3: Graph of Sitnikov motion for $z_0=0.2$ and $e=0.4$ ()

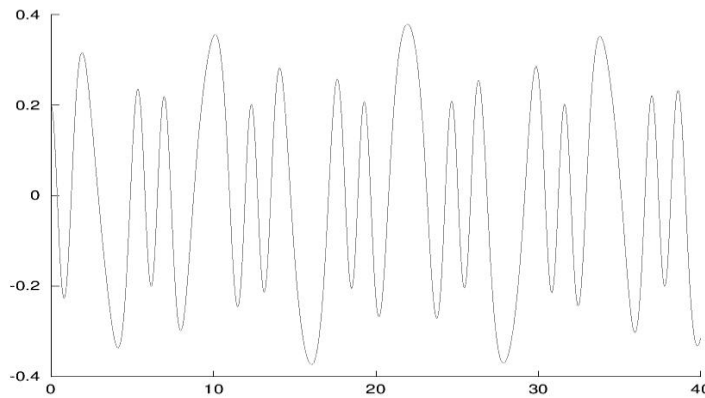


Fig 5.4: Graph of Sitnikov motion for $z_0 =0.2$ and $e=0$

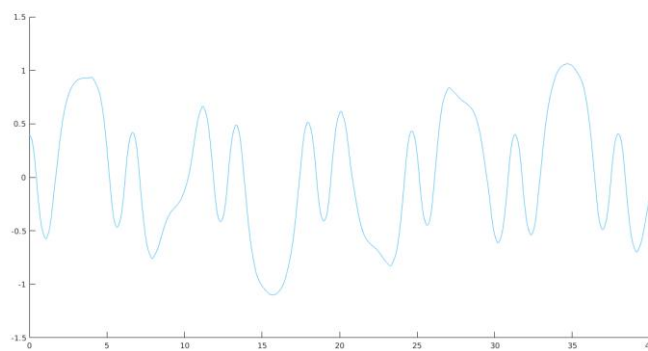


Fig 5.5: Graph of Sitnikov motion for $z_0=0.4$ and $e=0.6$

VI. DISCUSSION

The *Table 5.1* shows a close correspondence between the absorption bands obtained by HPC simulation with the corresponding experimental values with light variation. This result is truly realistic because theoretical calculation may not match precisely in this type of calculation with experimental value for so many factors e.g. the level of theory used (*UCAM-B3LYP*), selected basis sets (cc-pvtz) for the electron density which are limiting case or approximation of electronic structure models, solvent type, solvent concentration and the sample itself etc. Higher level of approximation should give closer result but take large computational power and time than that of lower level of approximation with less computational power.

Our focus was to verify performance of the newly built HPC with simulation, instead of traditional benchmark method. In the *Table 5.2* along with the graph of *Fig 5.3*, We saw that if we increase the number of the parallel nodes or CPU-core then the calculation time reduces significantly for the same computational task. More over the time reducing trend in *Fig 5.3* shows a lower rate for higher number of CPU-core. The HPC-cluster is in initial stage and functioning as we expected. Since total performance of a HPC cluster depends on three main key factors: (i). Cluster Hardware (ii) Network interface hardware and topology and (iii) The simulation package algorithm, to find the optimal condition more bench making is required with different approach.

We did not take in to account about the network performance and latency for the lack of resource since 1GB network interface for 4 node is more than enough, however as the HPC system will scale up to 100 nodes or more study on network interface saturation point, latency and program algorithm efficiencies are required.

We also tried to regenerate the solution of Sitnikov problem through custom code using Octave language as a serial job in the HPC-cluster. The graphs of *Fig 5.3*, *Fig 5.4* and *Fig 5.5*: are identical with the analytical solution (*Fig 4.2*) and shows the oscillatory motion for different value of initial position of z_0 and eccentricity e . This confirms our numerical simulation and effectiveness of the performance of simulation.

VII. CONCLUSION

This paper addresses the advantages of developing a High performance Computer Cluster (HPC) for Department of Physis, SUST as the requirements of computational power is increasing day by day. To verify HPC's functionality and performance simulated result has been compared with corresponding experimental value. This HPC Cluster have great advantages and can be widely used to undertake theoretical predictions, and research,.

The benefits are as follows:

- **Economic Benefit:** In fact this High performance Cluster Computing systems is to much more cheaper than traditional supercomputers of the same performance.
- **Scalability:** The system is designed to be salable, ie, easy to maintain and the performance can be upgraded just by adding additional Compute nodes.
- **Reliability:** As the system contain so many independent nodes, failure of one or more node except the Master node, will not make the total system to be down, it will continue to work but obviously with downgraded performance.

Other faculty also can be benefited to perform large computational task like Statistical analysis, DNA analysis, Molecular Dynamic Simulation etc. This HPC system has also opened a great opportunity to develop more knowledge on Parallel computing and Computational physics.

Through this development work, core technology of developing a HPC has been achieved facing so many challenges and solving the issues. The team is now confidently capable of enhancing the system to a larger scale. We need on going maintenance and monitor process to keep the HPC system up and running. Support is required to run the system continuously and to scale to a large system.

ACKNOWLEDGMENTS

We acknowledge the participation and the inspiration of the Nonlinear Optical Research group in developing the HPC. At the same time the Department of Physics, Shahjalal University of Science and Technology, Sylhet deserves thanks for providing the required space and logistics for the laboratory. Ministry of Education, The Government of Bangladesh deserves special thanks for supporting the project under HEQEP for developing the system. We also acknowledge that further enhancement of the cluster system has been partially supported by project ID: 2019/1/28 SUST research center.

Appendix A

Hardware List

Considering the cost effective approach, some commodity off the shelf computer hardware has been used to deploy the Cluster computer system. For master node relatively a higher configuration has been selected than that of the Worker node.

Master Node:

- Processor : AMD 3.0 Ghz 8 core
- Motherboard MSI
- Ram 8 GB , DDR2
- Hard disk 1 TB SATA 7200 rpm for server
- 500 watt ATX Power supply

Worker Node: (8 unit)

- Processor: Intel xeon 2.4 ghz 4 core , 8 mega cache memory
- Motherboard Super micro server motherboard
- Ram 2 GB , DDR2
- Hard disk : Disk less
- 350 watt ATX Power supply.

Network Switch:

- 100/1000 Mbps Switch 24 port.

Server rack: Custom Built

Operating System: for the Master-node linux operating system ubuntu linux 64 bit 17.04 lts with linux kernel 4.10 has been used. Following mandatory services has been configured to allow the disk less Compute-node to boot from network:

- DHCP - Dynamic Host Configuration Protocol
- TFTP- Trivial File Transfer Protocol
- PXELINUX -Pre-Boot Execution
- NFS - Network File System
- MPI, the message passing interface, has been installed for both master node and Worker nodes with their dependency and tested with this OS.
- The SLURM, a standard package for Job Scheduler, is included for user management and job management
- Password less authentication between the nodes has been enabled. Host name on /etc/hosts for both master and Worker has been kept identical so that there are no null or duplicate host name exists.
- NTP (Network Time Protocol) server on master node and NTP client on chroot has been installed and configured which allows nodes to synchronize time over network precisely.
- Other common dependency programming language library.
- Ganglia – Remote monitoring Package
- SSH Server – Enabled for remote access for pre-specified IP's

Appendix B

Example Input File: For Gaussian 09 Parallel Job

Here in below one input file named “2n4c.com” for Gaussian run of L-Tyrosine excited state energy calculation, except the first two line the rest of the parameter was generated with the Gaussian 09 and Gausview 5. The first two lines specify the number of node and the number of core per node to be used.

2n4c.com file content:

```
-----  
%NProcLinda=2  
%NProcShared=4  
%mem=12MW  
%chk=  
# td=(nstates=10) ucam-b3lyp/cc-pvtz scrf=(iefpcm,solvent=water)  
pop=(nbo,savemixed) scf=qc maxdisk=4GB geom=connectivity  
zitter-opt-td-hf  
0 1  
O -1.43528100 -1.46302000 -1.27912900  
O 4.36511900 -0.40326900 -0.44184800
```

O	-3.11005900	-1.49146900	0.17692700
N	-3.43140400	1.09615700	0.10059000
C	-1.09224400	0.92694100	0.84202400
C	-2.07892300	0.57109300	-0.28113700
C	0.33852100	0.56711800	0.52541200
C	0.98931300	-0.53202400	1.19580200
C	1.07974900	1.34385300	-0.44831800
C	-2.22312300	-0.95506200	-0.49214400
C	2.35133700	-0.86976100	0.87217900
C	2.44935800	1.00795800	-0.74671500
C	3.07893000	-0.07984700	-0.07607800
H	-1.40624400	0.43568500	1.75621500
H	-1.15083500	1.99655800	1.02225400
H	-1.80547100	1.05492700	-1.20485600
H	0.45265400	-1.12022200	1.91654800
H	0.61811900	2.18026200	-0.93980500
H	-3.83759900	0.47839000	0.78375500
H	-4.05694400	1.10677000	-0.68533000
H	2.82714100	-1.70602100	1.35303000
H	3.00746200	1.58109600	-1.46417700
H	4.85720100	-0.72818200	0.29196700
H	-3.40139600	2.02808200	0.47251200
1	10	2.0	
2	13	1.0	23 1.0
3	10	2.0	
4	6	1.0	19 1.0 20 1.0 24 1.0
5	6	1.0	7 1.0 14 1.0 15 1.0
6	10	1.0	16 1.0
7	8	1.5	9 1.0
8	11	1.5	17 1.0
9	12	1.5	18 1.0
10			
11	13	1.5	21 1.0
12	13	1.5	22 1.0
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			

3.3 Job Submission:

The job has been submitted remotely through the job submission tool SLURM and the .sh script for Gaussian job on 2 nodes is 2n4c.sh

2n4c.sh file content:

```
-----
#!/bin/bash
#!/bin/bash
#SBATCH -J g09-linda
#SBATCH -p y_comp -n 8
#SBATCH -A 2n4c.com
#SBATCH --output=2n4c.out
export OMP_NUM_THREADS=1
START_TIME=$SECONDS
g09 2n4c.com 2n4c.out
ELAPSED_TIME=$((SECONDS - START_TIME))
echo "Duration: $ELAPSED_TIME seconds"
#!/bin/bash
#SBATCH --job-name=g09-linda   ### jobname
#SBATCH --partition=y_comp     ### partition
#SBATCH --nodes=2             ### Number of nodes to use
#SBATCH --ntasks-per-node=1   ### Number of tasks per node (aka MPI processes)
#SBATCH --cpus-per-task=4     ### Number of cpus per task (aka OpenMP threads)
#SBATCH --output=2n4c.out     ### slurm output file
# Name of your input file ie 2n4c.com
JobFile=2n4c
# This creates a list of nodes that you job received to run on
LindaList=/nodes_linda.$SLURM_JOBID
touch $LindaList
# This creates a jobfile
JobName=./${JobFile}${SLURM_JOBID}.com
touch $JobName
# Create a list of hostnames and save it to the LindaList machine file
srun hostname -s | sort -u > $LindaList
# Tell linda to use ssh
export GAUSS_LFLAGS='-opt "Tsnet.Node.lindarsharg: ssh"'
# Read the contents of the machine file and put it in the job file
workers="%LindaWorkers="$(cat $LindaList | tr "\n" " " | sed "s/,$/")
# Write that out to the job file
cat <(echo "${workers}") ./${JobFile}.com > $JobName
START_TIME=$SECONDS
g09 ${JobFile}${SLURM_JOBID}.com ${JobFile}${SLURM_JOBID}.out
ELAPSED_TIME=$((SECONDS - START_TIME))
echo "Duration: $ELAPSED_TIME seconds"
echo "Job Complete"
```

Appendix C

Files to generate graph of Sitnikov solution

File : epsilon1.m

```
function [out] = epsilon1(omega,t,e)
out =
(3*omega+1)*t+(48/31)*e*sin(t)+(144/31)*e*omega*sin(t)+((3*omega+1)/2*omega)*sin(2*omega*t+(96/31)*
e*omega*sin(t));
return
end
```

File: epsilon2.m

```
function [out] = epsilon2(omega,t,e)
out = (3*omega-1)*t-(48/31)*e*sin(t)+(144/31)*e*omega*sin(t)+((3*omega-
1)/2*omega)*sin(2*omega*t+(96/31)*e*omega*sin(t));
return
end
```

File: epsilon3.m

```
function [out] = epsilon3(omega,t,e)
out =
(5*omega+1)*t+(48/31)*e*sin(t)+(144/31)*e*omega*sin(t)+((5*omega+1)/2*omega)*sin(2*omega*t+(96/31)*
e*omega*sin(t));
return
end
```

File: epsilon4.m

```
function [out] = epsilon4(omega,t,e)
out = (5*omega-1)*t-(48/31)*e*sin(t)+(144/31)*e*omega*sin(t)+((5*omega-
1)/2*omega)*sin(2*omega*t+(96/31)*e*omega*sin(t));
return
end
```

File: Z.m

```
function [out] = Z(z_o,omega,beta,gamma,t,e)
o_mul= z_o * (1-(24/31) * e * cos(t))/(1-(24/31) * e);
imul1=(1+(3/16) * z_o^2) * cos(omega * t + (48/31) * e * omega * sin(t));
imul2=(3/16)*z_o^2*cos(3*omega*t+(144/31)*e*omega*sin(t));
imul3= beta*e*(1-(1/2)*cos(2*omega*t+(96/31)*e*omega*sin(t)));
eps1= (5*z_o^2*sqrt(8)/(32*((3*omega+1)^2-gamma)))*cos(epsilon1(omega,t,e));
eps2= (5*z_o^2*sqrt(8)/(32*((3*omega-1)^2-gamma)))*cos(epsilon2(omega,t,e));
eps3= (3*z_o^2*sqrt(8)/(32*((5*omega+1)^2-gamma)))*cos(epsilon3(omega,t,e));
eps4= (3*z_o^2*sqrt(8)/(32*((5*omega-1)^2-gamma)))*cos(epsilon4(omega,t,e));
const1=(5*z_o^2*sqrt(8)/32)*((1/((3*omega+1)^2-gamma))+1/((3*omega-1)^2-gamma));
const2=(3*z_o^2*sqrt(8)/32)*((1/((5*omega+1)^2-gamma))+1/((5*omega-1)^2-gamma));
out = o_mul*(imul1-imul2+imul3*(eps1+eps2+eps3+eps4-const1-const2));
return
end
```

File: sitnicov.m

```
e = 0.4;
z_o = 0.2;
A = z_o*nthroot(8,4);
w_r=sqrt(8);
alpha = 6*sqrt(8);
beta = (66/31)*sqrt(8);
gamma=w_r^2-(3/2)*alpha*A^2;
delta = (3/2)*alpha*A^2;
omega = w_r-(9/4)*A^2;
t = 0:.02:40;
z_len = length(t);
```

```
z = zeros(z_len);  
for i=1:z_len  
    z(i,:)=Z(z_o.omega.beta.gamma.t(i).e);  
end  
plot(t,z,'DisplayName','z')
```

Command to run the calculation: octave > sitnicov.m

REFERENCES

- [1]. Eijkhout, V., Chow, E., & Geijn, R. v. (2011). Introduction to High Performance Scientific Computing (1st Edition ed.). Texas: The Saylor Foundation.
- [2]. Becker, D. J., Sterling, T., Savarese, D., Dorband, J. E., Ranawak, U. A., & Parker, C. V. (1995). Beowulf: A Parallel Workstation for Scientific Computation. International Conference on Parallel Processing. Stavrakas, I. K. (2005). Beowulf Clusters for Parallel Programming Courses. The International Conference on Computer as a Tool. 1, pp. 791-794. IEEE.
- [3]. Ayanda, D. & Adejumo (2011). A Prototype Model of High Performance Computing Using Beowulf Cluster. International Journal of Emerging Sciences, 1 (4), 696-705.
- [4]. Gaussian 09, Revision A.02, M. J. Frisch et al, Gaussian Inc., Wallingford, CT 200, 2009, 28.
- [5]. T. H. Dunning, Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. J. Chem. Phys. 90, 1989, 1007.
- [6]. Yanai, D. P. Tew, N. C. Handy, A new hybrid exchange - correlation functional using the Coulomb-attenuating method (CAM-B3LYP). Chem. Phys. Lett. 393, 2004, 51 - 57.
- [7]. Sitnikov, K. A.: 1960, 'Existence of oscillating motion for the three-body problem', Dokl. Akad. Nauk. USSR 133, 303 - 306.
- [8]. Faruque, S. B.: 2003, "Solution of the Sitnikov problem." Celestial Mechanics and Dynamical Astronomy 87, 353-369.
- [9]. Hagel, J.: 1992, A new analytical approach to the Sitnikov problem. Celestial Mechanics and Dynamical Astronomy 53, 267-292.
- [10]. Jeff Osborn, Internet Systems Consortium, Inc : 1994. Dynamic Host Configuration Protocol (DHCP) <https://www.isc.org/dhcp/>
- [11]. Karen R. Sollins (1980-01-29). The Trivial File Transfer Protocol (TFTP) Protocol. IETF. IEN 133. Retrieved 2010-05-01. <https://linux.die.net/man/1/tftp>
- [12]. H. Peter Anvin : 2008 Pre-Execution Environment-SYSLINUX for network Boot. <https://www.syslinux.org/old/pxe.php>
- [13]. Russel Sandberg, Bob Lyon, Bill Joy, Steve Kleiman, Sun Microsystems : 1989 Network File System (NFS) https://wiki.linux-nfs.org/wiki/index.php/Main_Page [14] William Douglas G. 2001 Message Passing Interface (MPICH) www.mcs.anl.gov/research/projects/mpich2/
- [14]. Morris Jette , Danny Auble : 2010 Simple Linux Utility for Resource Management (SLURM) <https://slurm.schedmd.com/>
- [15]. K. Feraud, M. S. Dunn, J. Kaplan, Spectroscopic Investigations of Amino Acids and Amino Acid Derivatives. I. Ultra-Violet Absorption Spectra of l-Tyrosine, dl-Phenylalanine, and l-Tryptophan. J. Biol. Chem. 112.1,1935,323-328
- [16]. S. A. Tarek, S. B. Faruque, S. M. Sharafuddin, H. Ara, and Y. Haque, 2020 UV-Visible absorption spectroscopy and Z-scan analysis along with corresponding molecular electronic structure analysis at DFT level for L-Tyrosine, IOSR J. Appl. Chem. 13, 45 - 51